

Modeling Customer Experience in a Contact Center through Process Log Mining

TENG FU, TeleWare, UK

GUIDO ZAMPIERI, Department of Computer Science & Information Systems, Teesside University, UK

DAVID HODGSON, TeleWare, UK

CLAUDIO ANGIONE, Department of Computer Science & Information Systems, Teesside University, UK

YIFENG ZENG, Department of Computer & Information Sciences, Northumbria University, UK

The use of data mining and modeling methods in service industry is a promising avenue for optimizing current processes in a targeted manner, ultimately reducing costs and improving customer experience. However, the introduction of such tools in already established pipelines often must adapt to the way data is sampled and to its content. In this study, we tackle the challenge of characterizing and predicting customer experience having available only process log data with time-stamp information, without any ground truth feedback from the customers. As a case study, we consider the context of a contact center managed by TeleWare and analyze phone call logs relative to a two months span. We develop an approach to interpret the phone call process events registered in the logs and infer concrete points of improvement in the service management. Our approach is based on latent tree modeling and multi-class Naïve Bayes classification, which jointly allow us to infer a spectrum of customer experiences and test their predictability based on the current data sampling strategy. Moreover, such approach can overcome limitations in customer feedback collection and sharing across organizations, thus having wide applicability and being complementary to tools relying on more heavily-constrained data.

CCS Concepts: • **Information systems** → **Data mining**; • **General and reference** → *Estimation*; • **Applied computing** → *Operations research*; Psychology;

Additional Key Words and Phrases: Customer experience, process log data, latent tree model, contact center

ACM Reference format:

Teng Fu, Guido Zampieri, David Hodgson, Claudio Angione, and Yifeng Zeng. 2021. Modeling Customer Experience in a Contact Center through Process Log Mining. 12, 4, Article 48 (July 2021), 21 pages. <https://doi.org/10.1145/3468269>

1 INTRODUCTION

The high competition in present business systems requires efficient and adaptive customer relationship management (CRM) strategies that are able to cope with the challenges of the current information era. Customer experience is today recognized as a dynamic and complex phenomenon, also thanks to the hugely increased rate of information exchange worldwide [24]. Customers' perceptions and evaluations indeed change over time and are influenced by a variety of factors, including the experience of other customers. Organizations are therefore increasingly interested

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2021 Copyright held by the owner/author(s). Publication rights licensed to Association for Computing Machinery.

XXXX-XXXX/2021/7-ART1 \$15.00

<https://doi.org/10.1145/3468269>

in the implementation of analytical tools that can routinely monitor their interactions with the customers and predict their future trends.

Technological development made possible the collection of large amounts of information, opening the door to quantitative CRM data science [14, 21]. In this context, contact centers are one of the main modality for customer-to-organization interactions and thus hold a key importance for CRM. At the same time, customer experience is becoming a core reference for companies and service providers, yet its prediction and interpretation is still a complex problem [35]. Such complexity arises from the multiple factors that jointly impact the customer experience, such as the dialog with the agent or the communication capacity of the contact center, and to the varying importance of such factors from organization to organization. For this reason, artificial intelligence is regarded as an essential component for process maintenance and optimization in contact centers, with great emphasis on natural language modeling [18]. As a result, recent research has widely focused on oral and written conversations between customers and agents [4, 7, 23, 28, 37, 38].

However, although spoken and written language analysis is highly attractive for organizations at large, in practice they often have to face many challenges before considering this tool, including resource, legislative and organizational barriers. For instance, growing concerns over privacy protection may represent a serious obstacle to data collection and management even for single organizations. Even when direct customer data (i.e. voice recording or feedback) can be collected and analyzed by an organization, middleware service providers can only access this information at discretion of the client organization. Moreover, from an individual caller's point of view, the feedback he/she provides mostly reflects the service enjoyed with the call center of the service provider (i.e. how professional the call center agent handles his/her problem), not the service quality from the infrastructure service provider. On the other hand, while direct customer feedback is increasingly being used as an indicator of experience quality, there exist also indirect forms of feedback that can be gained from data. Thus, in principle it is possible to infer the degree of satisfaction even if this is not directly provided by the customer. For instance, log-based process performance analysis has recently received impulse through various studies that characterized different aspects of process performance, also in relation to external factors and collaborative processes [27].

In this context, process logs represent an attractive resource, as they are an ubiquitous output of any communication channel and may often be the only data available for systematic investigations. Even if this type of data provides an indirect measurement of customer experience, its large-scale availability could allow infrastructure service providers to deploy informative systems which could better support the analytics strategy of client organizations. By offering domain knowledge injected and explainable data-driven models, end service providers could utilize those models as a supporting component of their own data analytics system, thus maximizing the value of contact center data. However, log-based modeling of customer experience remains largely unexplored.

The central objective of this study was to develop a methodology for interpreting log data of contact centers and for predicting associated customer experience. The principal challenge was the lack of knowledge on the true satisfaction related to available process realizations, which hinders direct supervised analysis. The final goal was to understand how customer experience information can be collected and managed across touch points and over time. In particular, we focused on log data of telephone calls, which remain the primary form of communication in contact centers [35]. Together, these elements constitute an unexplored setting where traditional approaches are not directly applicable. To tackle the problem, we thus combined unsupervised and supervised data mining and modeling techniques, seeking complementary indicators to formulate a consistent final interpretation. Our approach is based on a middleware service provider standpoint, which

enabled us to collect a large number of customer interactions from clients. Such scores have the potential to give an overall picture of the customer experience being delivered, both in real-time and historically. Also, from an application point of view, such analysis can allow service providers to perform inference on the underlying layers of the data, to suggest where customer experience can be improved, either for middleware service providers or individual organizations.

Our work reveals that popular data mining methods such as k -means clustering and elastic net fail to provide meaningful interpretations of log-based customer experience. On the other hand, application of latent tree modeling allowed us to characterize the log process in terms of different aspects of customer experience, here termed *facets* [31]. Such a modeling framework was indeed chosen due to its interpretative advantages and its elevated flexibility compared to ordinary Bayesian networks. By reconstructing and linking together latent facets in the log data, we could identify concrete points of intervention for improving customer experience. Moreover, we verified the predictability of the different aspects of customer experience via a multi-class supervised approach, exploiting extracted facet qualities as target labels. In summary, our contributions are the following:

- We built and validated a methodology for log-based modeling of customer experience that can cope with the absence of ground truth feedback from the customers.
- We demonstrate the utility of log data analysis and modeling in supporting contact center management, suggesting that practical advantages could be obtained through the deployment of online prediction tools as a component of existing call management systems.
- From an application point of view, we introduce an approach that can be implemented from middleware service providers, is widely applicable to a variety of contact center settings and can integrate client service providers' tools.

The article is organized as follows. In Section 2 we outline previous studies that investigate customer experience by means of data mining and modeling approaches. In Section 3 we outline the approach proposed in this work and describe the computational techniques used, while experimental outcomes are presented and discussed in Section 4. Lastly, in Section 5 we recapitulate the achievements and the significance of our work for CRM research.

2 RELATED WORK

A vast literature exists focusing on data mining strategies for CRM, dedicated to identify, attract, maintain and develop customers, with the largest fraction targeted on customer retention [19, 21, 29]. In contact centers, communication can take place through a range of channels such as telephone, e-mail, forums or online live chats. Phone call conversations constitute the main source of information, and approaches for customer experience prediction based on those have been recently investigated. Most studies concentrate on automatic customer satisfaction analysis, emotion detection, problematic call detection, and call segmentation [35]. Unstructured text or conversation data present the additional challenge of being converted to processable information. To that end, embedding semantic knowledge in sequential word data based on information gain has been shown to provide competitive and interpretable models while alleviating the burden of data preparation [7]. On the other hand, raw conversation data were successfully used to predict customer satisfaction by means of artificial neural networks [37]. More recently, the same task was investigated by combining entire phone call conversations and turn-level features [4, 23]. The ideal end stage for all these methods is to find application in end-to-end systems [33]. In parallel, phone call data was used to automate and optimize other tasks including customer recognition [34], customer routing [17], adequately pairing customer and agent based on historical data [26] and evaluating the work

carried out by agents [15]. In other types of channels data mining has been studied for instance in the context of chat routing modeling [20], emotion prediction from chats [28] and video chats [38].

However, conversation data may not be available depending on the organization. In some cases, only call logs may be provided - in the form of time-stamp entries - as contact centers are always supported by IT systems that log their execution. In these cases, contact center tasks and goals cross with process mining techniques. These have been used to evaluate time consumption of the process stages like specific activities or waits, to gain insights in resource allocation and workload management or to assess the service quality [27].

A further possibility is to combine heterogeneous data deriving from different communication channels, when available. As an example, Bockhorst *et al* have recently developed a comprehensive system that aggregates heterogeneous data sources such as dialog transcriptions, call logs, client and policy data in order to predict customer satisfaction [6]. The study could also benefit from self-reported satisfaction scores obtained from a third-party company.

In general, all of the previous approaches for predicting customer experience require ground truth feedback to be evaluated and implemented. In the frequent scenario where this is not available, a principled approach could be adopted. For instance, Fiedler *et al* proposed a general relationship linking quality of experience and quality of service in communication services [11]. However, this method overlooks all the specific aspects of the communication process, which are often critical for its optimization. To the best of our knowledge, no study has addressed data-driven modeling of customer experience starting uniquely from log data, which is the challenge targeted here. The settings of our case study are thus distinct from previous works and as such required the development of novel solutions, which are described in the following.

3 METHODS

In this section, we describe the ideas underlying our work and the computational techniques composing our customer experience modeling approach, which include a specific type of latent tree modeling and Naïve Bayes classification. Next, in Section 3.5 and 3.6 we briefly describe *k*-means clustering and elastic net regression, used as benchmark methods.

3.1 Proposed approach

The central goal of this work is to develop an approach to predict the quality of customer experience in a contact center starting uniquely from phone call logs. In ideal settings, organizations can directly or indirectly sample satisfaction rates to improve their services, possibly through proposed data mining techniques (see Section 2). We instead concentrated on the situation where no customer feedback is obtainable and investigated how to infer customer experience from available log-based data. This task can potentially be applied to a number of scenarios, yet it is generally overlooked in the literature. Here, we concentrated on the case of log data generated in a contact center during phone call processes.

We elaborated a strategy that could recognize and describe the various sub-processes underlying phone calls, in order to extract hidden quality levels of customer experience from the data itself. The central step here concerns the construction of a pouch latent tree model (PLTM) [31] of phone calls in terms of their temporal properties. Rather than identifying a global clustering across all input (manifest) variables, a PLTM embeds alternative data clustering solutions associated to some disjoint subsets. As explained in detail in Section 3.2, each of these clustering variants is represented by a discrete *latent variable* and defines a Gaussian mixture model (GMM) over its subset of manifest variables. Within a PLTM, manifest and latent variables are in conditional relationship based on a Bayesian tree among internal and terminal nodes. A PLTM thus provides alternative data groupings

and their probabilistic relationship, reflecting multiple data partitions examined at the same time. In other words, this model identifies different *facets* that jointly characterize a dataset, allowing the domain expert to evaluate all solutions provided. By considering complementary data facets, PLTMs therefore provide significant margin for expert interpretation and evaluation. In the case of process logs, some facets may indeed expose issues or critical points hidden across multiple sub-processes. Additionally, PLTMs have enhanced flexibility compared to ordinary Bayesian networks, as they can support real-valued input features such as event durations.

Upon reconstructing the classes defining the spectrum of all the likely customer experience quality levels, it is possible to build classification models that predict such classes within a supervised framework. The PLTM - combined with expert knowledge about the business domain - thus constitutes a tool to inform classification models and overcome the lack of feedback from the customer. In this second step, we considered the Naïve Bayes classifier, which is suited to a multiclass classification task with multiple customer quality levels, is computationally efficient and provides probabilistic confidence. Given these properties, this kind of model could be reasonably suited for implementation in a real-time call management system and was hence implemented in our approach.

3.2 Pouch latent tree modeling

A PLTM is a rooted tree model where leaf nodes represent input (or manifest) variables and internal nodes symbolize latent variables underlying the observable data distribution [31]. Manifest variables \mathbf{X} can either have discrete or continuous values \mathbf{x} , while all latent variables \mathbf{Y} are assumed to be discrete and represent data cluster labels \mathbf{y} . Links in the tree represent conditional dependencies of each variable on its parent. Additionally, leaf nodes can contain more than one input variable and are therefore referred to as *pouch nodes*. The latent variable connected to each pouch thus identifies a data partition based on the associated manifest variables.

PLTMs are strictly related to both GMMs and Gaussian Bayesian networks. On the one hand, graphical relationships in a PLTM are almost identical to those of a Gaussian network, with the only difference being that multiple manifest variables are allowed in a single node. On the other hand, PLTMs generalize GMMs in that they incorporate multiple latent variables and their probabilistic relationships. In a GMM, the input variables \mathbf{x} have the following distribution conditioned to cluster values \mathbf{y} :

$$P(\mathbf{x}) = \sum_{\mathbf{y}} P(\mathbf{y})P(\mathbf{x}|\mathbf{y}) \quad (1)$$

In a PLTM every pouch node is mapped to a disjoint subset of input features. Let us denote with $\Pi(Y)$ and $\pi(Y)$ the parent of a variable V and its value, respectively, and with \mathbf{W} and \mathbf{w} the variables inside a pouch and their values, respectively. The dependency of any discrete latent variable Y on its parent $\Pi(Y)$ is described by a conditional discrete distribution $P(\mathbf{y}|\pi(Y))$, whereas manifest variables \mathbf{W} follow a conditional Gaussian distribution $P(\mathbf{w}|\mathbf{y}) = \mathcal{N}(\mathbf{w}|\mu_{\mathbf{y}}, \Sigma_{\mathbf{y}})$ with mean $\mu_{\mathbf{y}}$ and covariance matrix $\Sigma_{\mathbf{y}}$. The probability distribution over manifest variables defined by a PLTM is

therefore the following:

$$\begin{aligned}
 P(\mathbf{x}) &= \sum_{\mathbf{y}} P(\mathbf{x}, \mathbf{y}) \\
 &= \sum_{\mathbf{y}} \prod_{j=1}^l P(y_j | \pi(Y_j)) \prod_{i=1}^b \mathcal{N}(\mathbf{w}_i | \mu_{\pi(\mathbf{w}_i)}, \Sigma_{\pi(\mathbf{w}_i)}) \\
 &= \sum_{\mathbf{y}} P(\mathbf{y}) \mathcal{N}(\mathbf{x} | \mu_{\mathbf{y}}, \Sigma_{\mathbf{y}}). \tag{2}
 \end{aligned}$$

We can thus see that a PLTM is a GMM, but it also provides richer information. First, it embeds alternative data clusterings based on different subsets of the input variables and identifies *facets* of the dataset. Second, it describes conditional dependencies among these facets.

Data clustering based on PLTM entails two main steps: estimating a set of parameters θ and learning a tree structure m . The former task can be performed by computing the maximum likelihood estimate θ^* through the expectation maximization algorithm [9], assuming a model structure m . The optimal PLTM structure m^* is learned by maximizing the Bayesian information criterion (BIC), which is defined as follows [36]:

$$BIC(m|\mathbf{X}) = \ln P(\mathbf{X}|m, \theta^*) - \frac{d(m)}{2} \ln n, \tag{3}$$

where $d(m)$ is the number of independent parameters in m . In this equation, the first term encourages good data fitting, whereas the second term is a regularization term penalizing complex models. Model parameters and structure are determined via a hill-climbing algorithm that iteratively generates new candidate models until the BIC is no longer improved [31].

As introduced in Section 3.1, in order to maximize the value of a dataset lacking ground truth customer feedback, a model with high explanatory power is an ideal choice for our research purposes. Therefore, although other models such as artificial neural networks have recently gained popularity in the field, they were not considered in this work due to the difficulties associated with their interpretation. Graphical models such as Bayesian networks, which capture probabilistic relationships among variables, thus became the main focus of this study. Given the multi-faceted nature of customer experience, we prioritized methods based on their flexibility and suitability to multiple variable types, hence converging on PLTMs.

3.3 Sensitivity analysis

We evaluated the strength of the relationships inside the PLTM by means of a sensitivity analysis in its probabilistic tree. We used an approach designed for discrete Bayesian networks to determine how the probability of a variable of interest (the hypothesis variable) is affected by a variation in the value of a single parameter in the network [8]. The sensitivity is computed by taking the parameters of a second node as *evidence* based on the data. Given an hypothesis node h , an evidence node e and a node parameter θ , we assumed a linear sensitivity function as follows:

$$f(\theta) = P(h|e)(\theta) = \frac{P(h \wedge e)(\theta)}{P(e)(\theta)} = \frac{a \cdot \theta + b}{c \cdot \theta + d}. \tag{4}$$

The sensitivity thus describes linear changes in the output probability $P(h|e)$ for infinitesimal variations in the parameters θ . It is calculated as the derivative of the sensitivity function at the original parameter value, as follows:

$$f'(\theta) = \frac{a \cdot d - b \cdot c}{(c \cdot \theta + d)^2}. \tag{5}$$

We performed this analysis with and without assuming evidence based on PLTM parameters [2].

3.4 Naïve Bayes classification

The Naïve Bayes classifier (NBC) is a popular multi-class classification method that exploits a probabilistic framework and is based on Bayes' theorem [12]. Under the assumption that every pair of features in the training dataset are independent, it aims to maximize the conditional probability of any observed data sample \mathbf{x} based on its label assignment y . Formally, the goal is to solve the following problem:

$$\max_y P(y) \prod_{i=1}^d P(x_i|y), \quad (6)$$

where d is the number of features and the conditional probabilities $P(x_i|y)$ are assumed to be Gaussian with mean μ_y and standard deviation σ_y , as follows:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right). \quad (7)$$

Parameters μ_y and σ_y are computed via maximum likelihood estimation.

The main advantages for choosing this method over other models are its probabilistic output and the computational efficiency. Differently from linear modeling, the probabilistic framework utilized by NBC in learning predictive patterns might confer advantages in terms of implementation into a usable classification system. Secondly, NBC only requires a small amount of training data for necessary parameter estimation, which makes it faster compared to more sophisticated methods. Decoupling the class conditional feature dependencies in fact allows independently estimating them as one-dimensional distributions. For these reasons, we employed NBC for facet-based classification.

3.5 k -means clustering

Along with the methods presented above, which are part of the approach introduced in this work, we applied other machine learning methods in the attempt to extract useful insights from log data. The k -means approach is one of the most widespread in clustering analysis, thus we chose it as a baseline in the task of identifying customer experience classes.

Assuming to have a set of n observations $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$, the k -means clustering method aims to partition these observations into k ($\leq n$) disjoint sets $\mathcal{S} = \{S_1, S_2, \dots, S_k\}$ by minimizing the intra-cluster variance in the feature space [22]. The task is mathematically formulated as follows:

$$\min_{\mathcal{S}} \sum_{i=1}^k \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \mu_i\|^2. \quad (8)$$

Here, μ_i represents the mean coordinates of the observations in the set S_i . The problem is solved by means of the Elkan's algorithm [10] using the k -means++ initialization method [5].

3.6 Elastic net regression

Elastic net is a least squares method for feature selection defined on n observations $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ stored in the matrix \mathbf{X} and on a vector of associated targets $\mathbf{y} = \{y_1, y_2, \dots, y_n\}$ that uses a combination of L_1 and L_2 regularization [40]. The problem solved is the following:

$$\min_{\beta} \left(\frac{1}{2n} \|\mathbf{y} - \beta\mathbf{X}\|_2^2 + \alpha l \|\beta\|_1 + \frac{\alpha}{2} (1-l) \|\beta\|_2^2 \right), \quad (9)$$

where α and l are hyper-parameters controlling joint and relative contributions of the L_1 and L_2 penalties, while β is a vector of linear regression coefficients.

This technique generalizes the least absolute shrinkage and selection operator method and ridge regression. With respect to these methods, it allows more effective feature selection and takes into account the correlation among independent variables. This offers elastic net higher flexibility and stability compared to other linear models. For such reasons, it was selected in this work as a baseline model for predicting and characterizing early call termination.

4 RESULTS

In this section, we firstly describe the data used in our work, which are entirely sampled from a real contact center. Next, we report the results of our initial efforts to model and interpret customer experience by using popular data mining methods. In the following of Section 4, we present the experimental work supporting our proposed approach.

4.1 Data

In this work, we focused on data provided by TeleWare, a middleware service provider involved in telecommunication development that operates with organizations from various sectors [3]. Here, we considered a dataset relative to a single organization and to the period between October and November 2016. The original dataset comprises time-stamp log recordings for 8,353 phone calls relatively to: call start, queue start, customer dialing, occurred or missed agent answer and call termination. The full set of data entries included in the log are visible in Table 1.

The structure of phone calls here considered is displayed in Figure 1. As soon as a call reaches the internal system, it is processed and assigned to a queue. This step typically occurs within the order of seconds, but occasionally it requires a longer time. Once the call is queued by the system, the customer has to select the desired service by dialing its corresponding code. Next, the system proceeds with the connection to an agent, depending on the selected service. However, the connection may fail as a result of a missed answer by the agent. This can happen when the agent is already occupied in another call or does not answer. At this point, the system tries to connect the customer to another agent, and if necessary to a third one and so forth, until a connection gets successfully established. The actual conversation can then take place, until either the customer or the agent terminates the call. However, it may also happen that the customer ends the call before being able to initiate any conversation.

Table 1. Example extract of a call log provided by the system. At the occurrence of any event relating to a given call, the log is updated with the time point corresponding to that event.

Call ID	Call start	Queue start	Customer Dialing	Agent answer	Agent busy	Agent missed answer	Agent unobtained	Call abandonment	Call connection	QBlock exit	Call end
114275506	04:50.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
114275506	04:50.0	05:02.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
114275506	04:50.0	05:02.0	09:38.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
114275506	04:50.0	05:02.0	09:38.0	09:43.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
114275506	04:50.0	05:02.0	09:38.0	09:43.0	NaN	NaN	NaN	NaN	09:44.0	NaN	NaN
114275506	04:50.0	05:02.0	09:38.0	09:43.0	NaN	NaN	NaN	NaN	09:44.0	09:44.0	NaN
114275506	04:50.0	05:02.0	09:38.0	09:43.0	NaN	NaN	NaN	NaN	09:44.0	09:44.0	11:05.0

Throughout the whole process, the occurring of these events and the occurrence time are recorded and stored in a database. It is therefore possible to completely characterize each call in terms of time-stamp information. Importantly, no information about actual customer satisfaction or feedback is available, as it was not sampled by the client company. This represents the main challenge in our

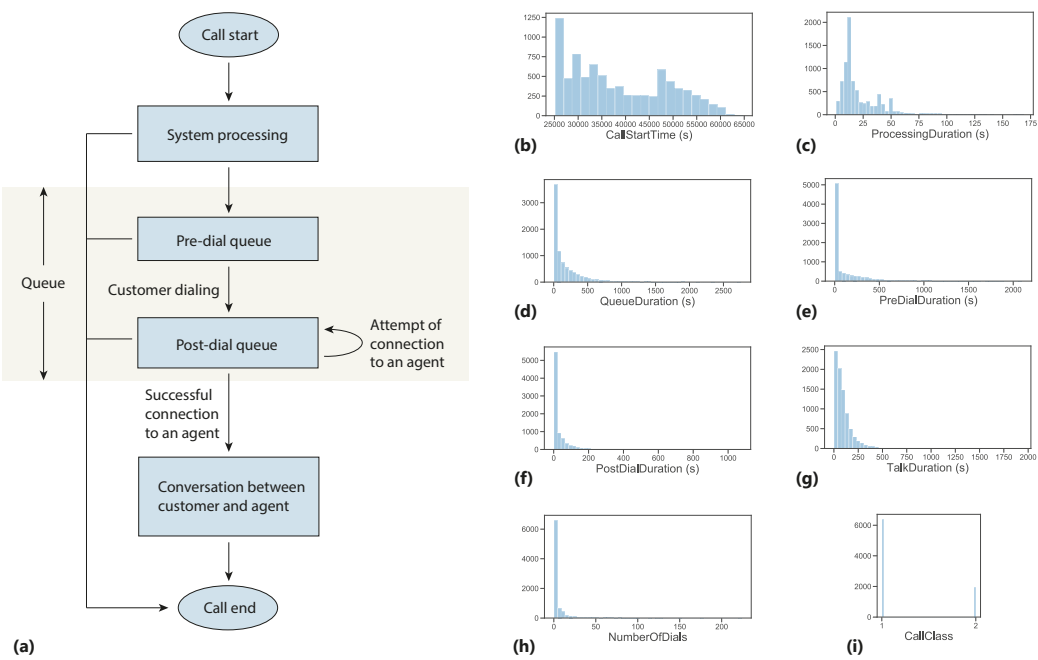


Fig. 1. Panel (a) illustrates the phone call schema for the considered contact center. Time points for each event in the boxes are automatically registered by the system as shown in Table 1 and comprise our initial dataset. Panels (b) to (i) show distributions for variables in Table 2 relative to the engineered dataset.

study, as it prevents the use of supervised data analysis techniques for directly predicting customer experience.

Table 2. Description of the features associated to phone calls used throughout the study, obtained upon transforming the original dataset. All variables are real except NumberOfDials and CallClass, which are integer.

Feature	Interpretation
CallStartTime	Instant of phone call connection to the system and start of its processing
ProcessingDuration	Time needed to the system to assign the incoming call to a queue
QueueDuration	Overall queue time
PreDialDuration	Queue time before the customer specifies the desired service
PostDialDuration	Queue time after the customer has specified the desired service
TalkDuration	Duration of the conversation between customer and agent
NumberOfDials	Number of times the system tries to connect the customer to an agent
CallClass	Categorical variable indicating whether the connection between customer and agent occurs before the end of the call (1 = yes, 2 = no)

In order to obtain a constant number of meaningful features for each sample, we engineered the original data, thereby obtaining the set of features displayed in Table 2 along with their explanation. A single variable - CallStartTime - remains in the form of time-stamp, which was

appropriately converted into a single numerical value expressed in seconds. Five other variables represent event durations within the call, corresponding to rectangular boxes in Figure 1. These are ProcessingDuration, QueueDuration, PreDialDuration, PostDialDuration and TalkDuration. Lastly, two further variables were introduced to characterize the events that may differ from call to call: the number of dials to an agent and whether or not the call terminates before an agent can respond. As a result, for instance, a log sequence like that displayed in Table 1 is converted into the process data sample shown in Table 3.

These features were used in all the subsequent analyses, described in the following sections. Unless differently stated, these were carried out in Python using the following packages: Pandas [25], Scikit-Learn [30], Matplotlib [16] and Seaborn [39].

Table 3. Example of data sample obtained upon feature transformation, corresponding to the log extract in Table 1.

CallID	CallStartTime	ProcessingDuration	QueueDuration	PreDialDuration	PostDialDuration	TalkDuration	NumberOfDials	CallClass
116539218	47091	12	282	276	6	81	1	1

4.2 Preliminary data exploration

Telecommunication, as a middleware service, is part of a network infrastructure, which means that the concepts and assumptions in network measurement can be usually applied to telecommunication. In a previous study, the authors proposed an intuitive hypothesis to map the general relationship between the quality of experience (QoE) and the quality of service (QoS) into a logarithmic relationship [11]. Adopted in a call center scenario, customer experience can be represented by the QoE while some features such as QueueDuration could represent the QoS. Embracing this idea, we initially formulated the following hypothesis: $CX = -a \cdot k \log(QD) + b$, where CX and QD symbolize customer experience and QueueDuration, respectively. By applying such a hypothesis to our dataset, we obtain a curve with parameters $a = 0.11$ and $b = 0.95$, which reflects the intuitive assumption that the longer a customer waits in the queue system and the worse his/her experience is. The obtained relation indicates this tendency in a reasonable way, but in practice, this hypothesis has the important drawback that it only focuses the global duration of the process, overlooking the relationships between multiple features within the dataset (i.e. QueueDuration and TalkDuration). For example, consider the case of a customer calling the credit card company for a card theft issue. The customer may wait in queue for a relatively longer time, but if the agent reacts quickly and professionally, the customer could also be satisfied. However, this scenario cannot be captured by the logarithmic model.

Next, we thus tried to identify phone call groups in our dataset that may be associated to different quality levels of customer experience through data mining algorithms that could leverage the detailed log-based information. The idea is that properties of the clusters may suggest novel indicators by association with factors known to positively or negatively influence customer satisfaction.

To this end, our dataset was partitioned by k -means clustering, one of the most popular cluster analysis methods [22]. We assessed the goodness of the partition for a range of possible number of clusters k , in terms of the silhouette coefficient, which expresses the degree of similarity among observations in a same cluster compared to observations in other clusters [32]. Let us indicate with $a(\mathbf{x})$ the average Euclidean distance between observation \mathbf{x} and all other samples in the same cluster and with $b(\mathbf{x})$ the smallest average distance to the samples in any other cluster. The silhouette

coefficient $s(\mathbf{x})$ is defined as:

$$s(\mathbf{x}) = \frac{b(\mathbf{x}) - a(\mathbf{x})}{\max\{a(\mathbf{x}), b(\mathbf{x})\}}, \quad (10)$$

and ranges in the interval $[-1, 1]$, with 1 being the maximal consistency with the clustering.

As it is possible to see in Figure 2, the average silhouette across all samples is considerably low for every value of k up to 20, with only $k = 2$ lying above 0.30. Visual inspection of silhouette profiles and data separation confirmed that the global quality of the clustering remains modest across all values of k . For instance, Figure 2 shows that for $k = 2$ a large cluster dominates over a smaller and less defined one, while Figure 3 displays the poor separation obtained.

Moreover, binary separation is not ideal for practical interpretation as customer experience hardly reduces to a sharp positive/negative response. The case with the second highest average silhouette, $k = 8$, would appear more meaningful, if the associated clustering quality did not remain so poor. Additionally, the obtained models do not incorporate any domain knowledge, producing difficulties in terms of interpretation. Overall, this analysis did not provide useful insights in the various forms of customer experience delivered.

As a second task, we sought to build a predictive model for early call termination to characterize the relationships with all previous process events. We employed elastic net, a regularized regression method that allows to estimate the correlation among variables [40]. In elastic net, the weight of each variable is controlled by both L_1 and L_2 penalties, whose relative contribution is balanced through two hyper-parameters α and l as shown in Equation 9. We thus built a series of elastic net models using CallClass as a target variable and studying the influence of different L_1 and L_2 ratios over an interval $[0.1, 1]$. We optimized the models by tuning the hyper-parameter α based on a 20-fold cross-validation and the mean squared error (MSE), defined as follows:

$$\text{MSE} = \frac{1}{n^{\text{test}}} \sum_{i=1}^{n^{\text{test}}} (y_i - \hat{y}_i)^2, \quad (11)$$

where y_i and \hat{y}_i are the predicted and true CallClass values for any observation i , respectively, while n^{test} is the number of test observations in each test data split. Figure 4 shows results for the

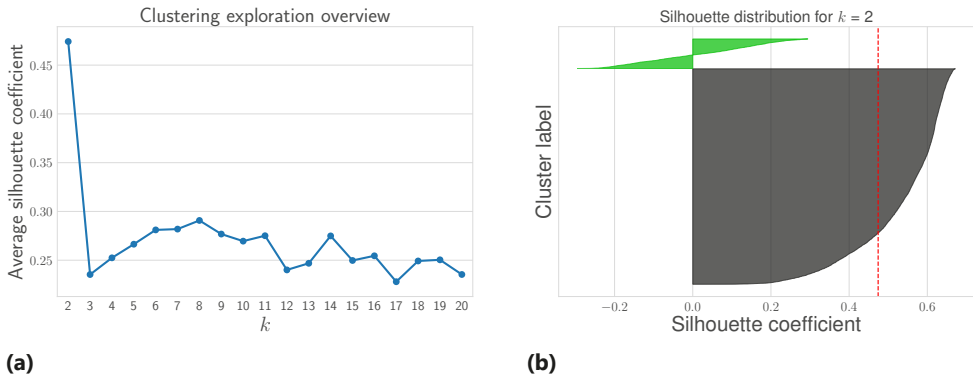


Fig. 2. Results of the k -means clustering analysis: **(a)** average silhouette index as a function of the number of clusters k ; **(b)** silhouette distribution corresponding to $k = 2$. The dashed red line here indicates the average silhouette index across all samples.

best-performing model, with a MSE of roughly 0.11. In panel **(b)** it is possible to view the associated model parameters, from which we can infer the variables most associated to early call termination.

As it could be expected, QueueingDuration is one the most correlated variables with CallClass. Moreover, QBeforeDialingDuration appears much more relevant than QAfterDialingDuration. This suggests that the early stage of the call is highly determinant on customer's willingness to continue the wait. Conversely, CallStartTime very weakly affects CallClass and NumberOfDials does not influence it at all. The moment of the day therefore does not emerge as a factor for premature call termination. Altogether, these results indicate that the service provider needs to improve the initial handling of the calls inside the system.

This analysis could thus reveal expected and unexpected patterns in the data, but is nevertheless limited in scope to the occurring of a conversation between customer and agent. Particularly, it missed the overall quality of service experienced and degree of satisfaction. We therefore moved on to a different approach, presented in the following section.

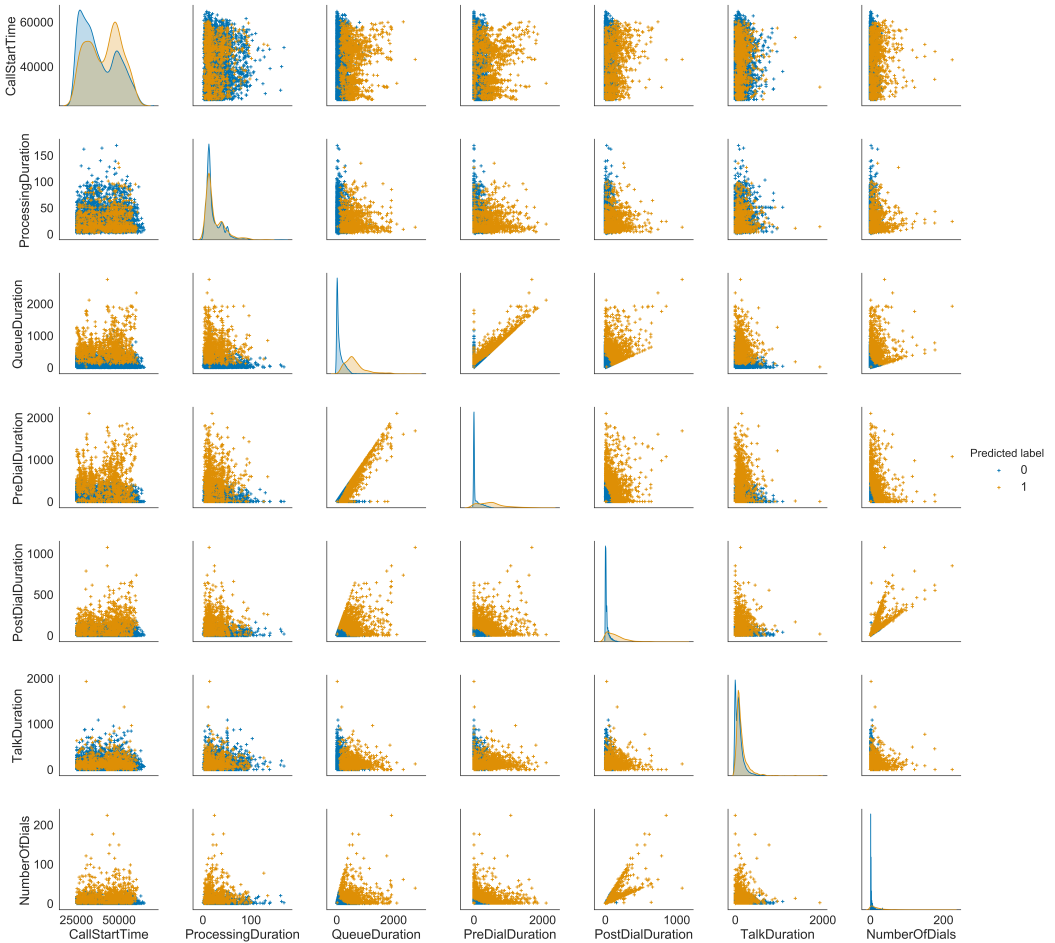


Fig. 3. Feature distributions and correlations in terms of distinct clusters identified by the k -means analysis with $k = 2$. Best viewed in color.

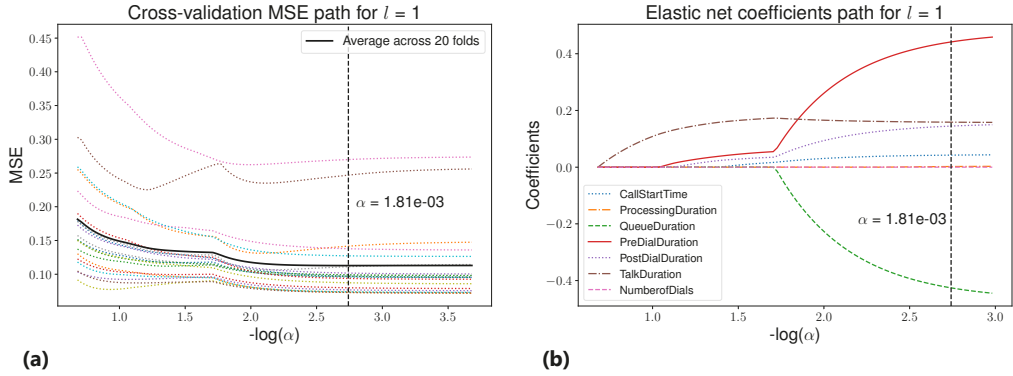


Fig. 4. Optimization of the elastic net model with CallClass as target variable, based on regression MSE. Panel (a) shows the MSE trend across varying values of the hyper-parameter α when $l = 1$. Panel (b) displays the trend of related model coefficients for all explanatory variables. The black vertical line indicates the final selected model. Best viewed in color.

4.3 Pouch latent tree modeling of log-based processes

To achieve a more meaningful process characterization, we next applied our proposed approach described in 3.1. We thus built a PLTM of our phone call log dataset in term of the variables described in Table 2, except CallClass that was dropped. In this process, we utilized the PLTM Java implementation made available by Poon *et al* [31] through the JPyPe interface [1]. The resulting structure is illustrated in Figure 5. As it is visible, four latent variables represent alternative clustering solutions, each one corresponding to a complementary data facet. Variable Y_3 is linked only to CallStartTime, so it represents a grouping of the data based on this feature alone. Its inner states are five, meaning that there are main intervals wherein the incoming calls are allocated. Variables Y_1 , Y_2 and Y_4 are assigned pouches with two manifest features and nine data clusters each.

Table 4 displays the mean of each input feature conditional on the assigned clustering variable. Inside Y_1 , state 0 emerges as the most numerous cluster and corresponds to very short global and pre-dial queue time spans. Along with it, five states of decreasing probability (in the order: 8, 1, 7, 5 and 6) describe increasingly long queue durations. Other three states identify waiting times under average (state 2) or long queue durations with very quick pre-dial times (states 3 and 4). The hidden variable Y_1 thus appear associated to the overall system efficiency in quickly providing information to the customer and handle his/her request.

Inside Y_2 , there is instead more robust consensus between high and low manifest variables values. Clusters are divided between states with post-dial queue time and number of dials above (states 0, 2, 3 and 7) and below (states 1, 4, 5 and 8) average. Only state 6 is characterized by a PostDialDuration slightly above than average and a NumberOfDials just below, which may highlight an anomalous duration for each connection attempt. Variable Y_2 is arguably related to customer patience or queue behavior, as its pouch contains features belonging to the end-stage queue.

Next, Y_3 is evidently related to the initiation time of the call, which does not result directly associable to any other feature. It is banally divided into clusters where the average CallStartTime is low - such as for states 0, 3 and 4 - or high like for states 1 and 2.

Lastly, Y_4 is more evenly distributed across clusters with different combinations of high and low feature values, as expected from their heterogeneous semantics. Two of the most likely states are

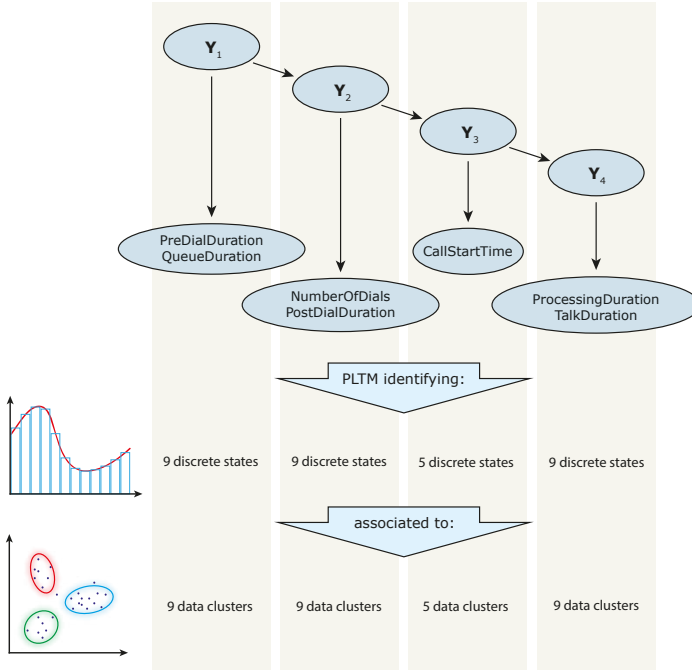


Fig. 5. Structure of the obtained phone call PLTM. Each non-terminal node corresponds to a discrete latent variable, representing a data clustering solution based on the manifest variables associated to its child pouch node. Relations among latent variables constitute a linear Bayesian tree broadly reflecting the temporal evolution of a phone call.

associated to low ProcessingDuration and TalkDuration (states 2 and 4), but clusters 3, 5, 6 and 8 present mixed tendencies. Such a data partition suggests that Y_4 may represent specific services provided by the company. For example, state 4 might be related to appointment scheduling while state 5 might concern information requests, as the latter has higher TalkDuration.

Overall the identified process facets appear reasonable, as queue-related attributes are grouped together. It is however interesting to notice that CallStartTime is alone in the pouch and that ProcessingDuration and TalkDuration are grouped together. The initiation time has therefore no evident correlation with queue behavior or dialog duration. Moreover, as clearly visible from Figure 5, the obtained PLTM has a flat structure where the latent component of the tree is represented by a chain. This indicates that simple relationships exist among the different components of any phone call. Reasonably, the sequence of latent variables largely reflects the temporal events during a call, with Y_1 being the first node followed by Y_2 and Y_4 as the last node. The PLTM structure thus supports its general soundness.

To evaluate the PLTM robustness and gain further insights from it, we applied a sensitivity estimation method for Bayesian networks [8] described in Section 3.3 and available on Bayes Server [2]. To this end, we transformed the PLTM in a purely discrete tree by pruning pouch nodes and leaving each latent node as a discrete variable whose states are associated to previously identified clusters. As a result, the new tree is a chain composed only of discrete latent variables, each one with a number of states equal to the number of clusters in the corresponding facet. Nodes Y_1 , Y_2 and Y_4 thus have nine states while Y_3 has five. Assuming linear sensitivity relationships as in

Table 4. Overview of the latent variables identified and their internal states. Columns relative to manifest attributes show the average within each cluster corresponding to a given internal state. The last row represents the average over the complete data distribution.

State	Y ₁			Y ₂		
	P(state)	QueueDuration	PreDialDuration	P(state)	PostDialDuration	NumberOfDials
0	0.54	33.1	7.8	0.02	84.0	21.8
1	0.06	377.3	283.7	0.11	19.7	2.2
2	0.07	113.8	23.4	0.16	229.1	18.9
3	0.04	237.3	8.5	0.07	94.5	8.1
4	0.01	500.1	2.4	0.20	0.1	2.0·10 ⁻²
5	0.05	734.4	708.1	0.18	8.1	1.0
6	0.01	1,317.6	954.6	0.08	36.3	3.6
7	0.02	544.9	306.7	0.06	331.4	55.5
8	0.20	247.3	231.6	0.13	6.4	1.0
Overall	1.00	173.5	120.0	1.00	35.3	4.3

State	Y ₃		Y ₄		TalkDuration
	P(state)	CallStartTime	P(state)	ProcessingDuration	
0	0.15	2.7·10 ⁴	0.04	58.1	467.0
1	0.09	4.6·10 ⁴	0.11	35.5	142.9
2	0.32	5.0·10 ⁴	0.03	10.6	4.0·10 ⁻²
3	0.14	2.8·10 ⁴	0.10	11.9	237.3
4	0.31	3.4·10 ⁴	0.22	12.2	61.3
5			0.24	12.3	114.3
6			0.15	38.0	4.6·10 ⁻²
7			0.02	55.1	235.2
8			0.10	33.0	65.3
Overall	1.00	3.9·10 ⁴	1.00	22.4	94.3

Equation 4, we estimated the impact of each latent variable on the others as by Equation 5. In this stage we concentrated on a one-way sensitivity analysis in which a single parameter is separately varied, and we considered two different types of perturbation: with and without evidence. Results are summarized in Table 5 and 6 respectively.

In the scenario where evidence is ignored, low average sensitivity values can be observed over all node combinations, maintaining at or below 0.14 among non-trivial pairs. These results thus indicate that the three main call components - call start time, queue phase and conversation/service characteristics - do not have strong influence on each other. The highest sensitivity was registered for Y₂ with respect to Y₁, consistently with the contribution for the total queue duration on PostDialDuration. Furthermore, despite its position in the tree TalkDuration is the most insensitive variable on average. This seems to be conflict with the result of elastic net feature selection in Section 4.2, but the PLTM interpretation makes sense in practical situations. In the real world, the conversation duration depends on the requested service, thus it would not be suitable to simply assign a sample with longer conversation time to a low customer satisfaction. Conversely, CallStartTime has relatively stronger sensitivity in terms of global and early queuing features, denoting a non-negligible - even if weak - relationships between them.

Table 5. Summary of node-to-node sensitivity analysis without evidence. Values indicate average, maximum and minimum sensitivity for the manifest variables associated to each latent variable.

Hypothesis node	Parameter Node	Avg	Max	Min
Y ₁	Y ₁	0.30	1	-0.67
	Y ₂	0	0	0
	Y ₃	0	0	0
	Y ₄	0	0	0
Y ₂	Y ₁	0.14	0.79	-0.27
	Y ₂	0.02	0.54	-0.28
	Y ₃	0	0	0
	Y ₄	0	0	0
Y ₃	Y ₁	0.11	0.41	-0.28
	Y ₂	0.01	0.33	-0.25
	Y ₃	0.05	0.24	-0.22
	Y ₄	0	0	0
Y ₄	Y ₁	0.05	0.46	-0.19
	Y ₂	$3.78 \cdot 10^{-3}$	0.36	-0.19
	Y ₃	$4.94 \cdot 10^{-3}$	0.18	-0.05
	Y ₄	0.04	0.32	-0.14

Furthermore, the obtained sensitivities display symmetry between positive and negative values among long and short duration states. This symmetry indicates that call stages have a tendency to distribute either around short durations (abandoned calls or quickly deliverable services) or long durations due to both complicated customer requests and system or agent inefficiency. In fact, long conversations are more associated to long queuing times as can be seen by the higher sensitivity value between Y₁ and Y₄ compared to the other variables (although the value is still low: 0.05).

In the case where evidence nodes are taken into account, global sensitivity increases although it remains moderately low. In particular, average sensitivity values are considerably higher for a few specific combinations of hypothesis, evidence and parameter shown in Table 6. For instance, variable Y₃ becomes more sensitive to Y₁ and Y₄ only when evidence is in Y₄. This suggests that specific services required by the customer may influence when he/she is going to make the call and how long he/she is willing to wait for the connection to an agent and for the fulfillment of his/her requests. Y₁ instead shows strong influence to other variables, regardless of which variable is set to be evidence node. This is probably due to Y₁ being considered as the root node of the structure. Finally, Y₄ only affects itself, independently of the evidence node. This suggests that the conversation span depends more consistently on external variables not considered here (booking, technical problems, etc.).

According to the results presented above, the main factors that affect customer experience include the following:

- (1) Call incoming time: related to latent variable Y₃.
- (2) Customers' patience and willingness while queuing: related to Y₁ and Y₂.
- (3) Services that are provided by the company: related to Y₄. Specific services may indeed intrinsically require more time.
- (4) Call center and system handling capacity: related to all latent variables.

Table 6. Summary of node-to-node sensitivity analysis with evidence, for a subset of hypothesis-parameter-evidence triplets with higher sensitivity. Values indicate average, maximum and minimum sensitivity for the manifest variables associated to each latent variable.

Hypothesis node	Parameter node	Evidence node	Avg	Min	Max
Y ₁	Y ₁	Y ₂	0.31	-11.88	25.62
		Y ₃	0.20	-3.39	3.83
		Y ₄	0.22	-3.15	3.68
Y ₂	Y ₁	Y ₃	0.10	-1.09	1.65
		Y ₄	0.14	-0.65	1.19
Y ₃	Y ₁	Y ₄	0.06	-0.34	0.42
	Y ₃	Y ₁	0.04	-0.88	0.88
		Y ₂	0.04	-1	1
	Y ₄	Y ₄	0.17	-10.27	11.78
Y ₄	Y ₄	Y ₁	0.05	-0.54	0.66
		Y ₂	0.06	-0.62	0.69
		Y ₃	0.04	-1	1

Clearly, factors on the customer side (1 and 2) are subject to intrinsic uncertainty and randomness, while factors from the company side (3 and 4) can be controlled and improved.

4.4 Facet-based classification of customer experience

In the perspective of having a predictive system used by the company, not only the customer experience has to be estimated correctly, but the estimate has to be provided quickly. For instance, the system could provide a response to contact center agents immediately after a call ends as a feedback on the delivered service. In such a case, predictive algorithms need to be efficient enough not delay the start of following tasks. However, computation of a PLTM is too expensive in these situations, since both parameters and structure have to be re-calculated at every change in the training set. Here we considered a simple process, but time can swiftly increase with the ramification of sub-processes. Firstly, the PLTM is determined using a greedy search strategy which computes every possible tree based on the seed structure that is generated in the previous iteration. The search algorithm performs operations like adding, removing and relocating states and nodes, thus recomputing all the associated parameters through a computationally-expensive expectation maximization procedure. Secondly, PLTM inference is based on the junction tree algorithm for Bayesian networks, which considers every possible clique in the tree structure, thus further increasing the algorithmic complexity. When the number of variables is large, the cost in terms of computation resources (CPU) and time can therefore be very high.

For these reasons, we tested the transformation of the PLTM clustering into a classification task with a more efficient algorithm. Building on results in the previous sections, we translated each of the clustering solutions given by the PLTM into labels for separate classification tasks. Every latent state for any given facet was thereby taken as a target class for the task relative to that facet. We then simulated customer satisfaction predictions within the existing queue and data collection system in a scenario where true labels are available. In this phase, we employed

Table 7. Summary of multi-class classification performance in the four scenarios defined by the PLTM facets. For each scenario, average accuracy and Matthews correlation coefficient R_K in 3-fold cross validation, as defined in Equation 12 and 13.

Facet	Accuracy	R_K
Y_1	$0.91 \pm 2.3 \cdot 10^{-3}$	$0.87 \pm 3.6 \cdot 10^{-3}$
Y_2	$0.90 \pm 1.2 \cdot 10^{-3}$	$0.88 \pm 1.4 \cdot 10^{-3}$
Y_3	$0.92 \pm 6.0 \cdot 10^{-3}$	$0.89 \pm 7.9 \cdot 10^{-3}$
Y_4	$0.92 \pm 6.9 \cdot 10^{-3}$	$0.90 \pm 8.0 \cdot 10^{-3}$

Naïve Bayes classifier (NBC) models under the assumption of independent features as explained in Section 3.4.

To evaluate the efficacy of each NBC, we employed a 3-fold cross-validation as follows. The entire dataset was split in three folds of equal size and with approximately the same ratios of class labels. In turn, two data folds (~67% of the overall dataset), were used for training while the remaining fold (~33% of the overall dataset) for testing and validation. We repeated this procedure three times, swapping the test fold at each iteration. In this way, we evaluated NBC models in the scenario relative to any clustering facet based on three test sets of roughly 2,800 samples each. Prediction performance was assessed based on popular measures for multi-class classification: accuracy and Matthews correlation coefficient. Assuming that n^{test} observations are classified into K classes, we indicate with C the confusion matrix where any entry C_{kl} represents the number of observations belonging to class k and assigned to class l by the model. In a multi-class scenario, the accuracy is defined as the global ratio of correctly assigned observations to the true classes, as follows:

$$\text{Accuracy} = \frac{1}{n^{test}} \sum_{k=1}^K C_{kk}. \quad (12)$$

The accuracy thus reflects the ratio of correct predictions disregarding the number and ratios of considered classes.

The Matthews correlation coefficient is instead a measurement of agreement between real and predicted values that takes into account the proportion of class cardinalities. Generalizing the binary label case, the multi-class Matthews coefficient R_K is defined as follows [13]:

$$R_K = \frac{\sum_{klm} C_{kk} C_{lm} - C_{kl} C_{mk}}{\sqrt{\sum_k (\sum_l C_{kl}) (\sum_{l', k' \neq k} C_{k'l'})} \sqrt{\sum_k (\sum_l C_{lk}) (\sum_{l', k' \neq k} C_{l'k'})}}. \quad (13)$$

Based on this equation, $R_K \in [-1, 1]$ and values close to 1 are obtained only if elements from all classes are recognized correctly. It is therefore a precise indicator of good prediction balancing the contribution of every misclassification possibility. Together, accuracy and R_K provide a complete overview of classification performance.

Overall, classification performances maintain high across all scenarios, as summarized in Table 7. Both accuracy and Matthews coefficient result steadily close to 0.90 across test sets with deviations within 1%. Figure 6 illustrates the global counts of correctly and incorrectly classified observations across all classes. Manual inspection of misclassified samples allowed us to verify that they belong to statistically similar clusters. These results therefore support the use of automatic prediction methods within the call management system, possibly as online classification tools.

5 CONCLUSIONS

In this study, we investigated how to leverage process log data to characterize and achieve insights into customer experience within a contact center. To develop and validate our methodology, we focused on the application of data mining and modeling techniques on real-world call center data obtained from the TeleWare database, consisting of log timing information for thousands of calls and lacking customer feedback on the service received. Since traditional clustering algorithms such as k -means clustering could not provide reasonable and meaningful interpretations, we used a PLTM to determine alternative data partitions reflecting complementary aspects. The obtained PLTM combined with domain knowledge allowed us to characterize the variation in log-based processes and hypothesize different degrees of customer experience. Subsequent classification experiments verified the efficient predictability of customer experience facets based on log data, potentially implementable as a support system.

Customer experience is a complex and multifaceted phenomenon which requires non-obvious strategies to be understood and managed. Our findings suggest that taking into account the different aspects in a certain process can provide more meaningful insights compared to mining process data as a whole. Moreover, given the wide and diversified range of business processes, it is essential to develop and validate computational techniques that can leverage the available data. Our results demonstrate that application of PLTM can be useful in contexts where customer self-reported

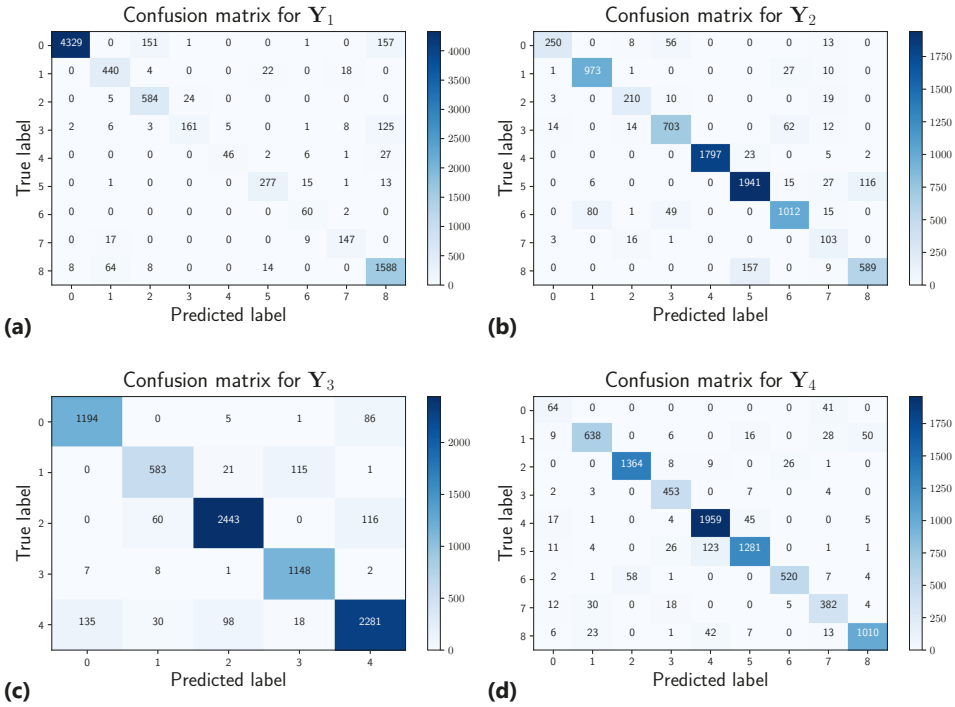


Fig. 6. Confusion matrices for NBC multi-class classification in 3-fold cross-validation, for each of the PLTM clustering scenarios: (a) Y_1 ; (b) Y_2 ; (c) Y_3 and (d) Y_4 .

feedback is not available to identify concrete points of improvement in CRM. We therefore envisage that our approach can be extended to other log-based processes.

Although the obtained results demonstrate the value of a PLTM in the considered industrial environment, there are issues yet to solve before considering it usable in a realistic setting. The main obstacle is the computational cost of the algorithms for PLTM construction. For this reason, we studied the possibility of performing customer experience prediction through a more efficient supervised method. Our results support the development of a two-stage strategy with a preliminary phase of facet labels identification and a real-time phase of multi-label classification. In future research we will concentrate on a redevelopment of PLTM to achieve a higher computation efficiency and to reduce the time cost in model training. To address this issue, we see artificial neural networks as a promising option.

ACKNOWLEDGMENTS

The work was supported by the Knowledge Transfer Partnership (KTP) KTP010406 by Innovate UK.

REFERENCES

- [1] 2009. JPyPe - Bridging the worlds of Java and Python. (2009). <http://jpype.sourceforge.net/>
- [2] 2017. Bayes Server - Bayesian network software for Artificial Intelligence. (2017). <https://www.bayesserver.com>
- [3] 2017. TeleWare plc. (2017). <https://www.teleware.com/>
- [4] Atsushi Ando, Ryo Masumura, Hosana Kamiyama, Satoshi Kobashikawa, and Yushi Aono. 2017. Hierarchical LSTMs with Joint Learning for Estimating Customer Satisfaction from Contact Center Calls. *Proc. Interspeech 2017* (2017), 1716–1720.
- [5] David Arthur and Sergei Vassilvitskii. 2007. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 1027–1035.
- [6] Joseph Bockhorst, Shi Yu, Luisa Polania, and Glenn Fung. 2017. Predicting Self-reported Customer Satisfaction of Interactions with a Corporate Call Center. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 179–190.
- [7] Andrea Brunello, Enrico Marzano, Angelo Montanari, and Guido Sciacicco. 2018. J48S: A Sequence Classification Approach to Text Analysis Based on Decision Trees. In *International Conference on Information and Software Technologies*. Springer, 240–256.
- [8] Enrique Castillo, José Manuel Gutiérrez, and Ali S Hadi. 1997. Sensitivity analysis in discrete Bayesian networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans* 27, 4 (1997), 412–423.
- [9] Arthur P Dempster, Nan M Laird, and Donald B Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)* (1977), 1–38.
- [10] Charles Elkan. 2003. Using the triangle inequality to accelerate k-means. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*. 147–153.
- [11] Markus Fiedler, Tobias Hossfeld, and Phuoc Tran-Gia. 2010. A generic quantitative relationship between quality of experience and quality of service. *IEEE Network* 24, 2 (2010), 36–41.
- [12] Nir Friedman, Dan Geiger, and Moises Goldszmidt. 1997. Bayesian network classifiers. *Machine learning* 29, 2-3 (1997), 131–163.
- [13] Jan Gorodkin. 2004. Comparing two K-category assignments by a K-category correlation coefficient. *Computational biology and chemistry* 28, 5-6 (2004), 367–374.
- [14] Jiawei Han, Jian Pei, and Micheline Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.
- [15] Hui-Huang Hsu, Te-Chan Chen, Wei-Tsung Chan, and Jung-Kuei Chang. 2016. Performance Evaluation of Call Center Agents by Neural Networks. In *Advanced Information Networking and Applications Workshops (WAINA), 2016 30th International Conference on*. IEEE, 964–968.
- [16] John D Hunter. 2007. Matplotlib: A 2D graphics environment. *Computing in science & engineering* 9, 3 (2007), 90–95.
- [17] Fernando Jiménez, Enrico Marzano, Gracia Sánchez, Guido Sciacicco, and Nicola Vitacolonna. 2015. Attribute selection via multi-objective evolutionary computation applied to multi-skill contact center data classification. In *Computational Intelligence, 2015 IEEE Symposium Series on*. IEEE, 488–495.
- [18] Keith Kirkpatrick. 2017. AI in contact centers. *Commun. ACM* 60, 8 (2017), 18–19.

- [19] Vineet Kumar and Werner Reinartz. 2018. *Customer relationship management: Concept, strategy, and tools*. Springer.
- [20] Benjamin Legros and Oualid Jouini. 2018. On the scheduling of operations in a chat contact center. *European Journal of Operational Research* (2018).
- [21] Gordon S Linooff and Michael JA Berry. 2011. *Data mining techniques: for marketing, sales, and customer relationship management*. John Wiley & Sons.
- [22] Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE transactions on information theory* 28, 2 (1982), 129–137.
- [23] Jordi Luque, Carlos Segura, Ariadna Sánchez, Martí Umbert, and Luis Angel Galindo. 2017. The role of linguistic and prosodic cues on the prediction of self-reported satisfaction in contact centre phone calls. *Proc. Interspeech 2017* (2017), 2346–2350.
- [24] Janet R McColl-Kennedy, Anders Gustafsson, Elina Jaakkola, Phil Klaus, Zoe Jane Radnor, Helen Perks, and Margareta Friman. 2015. Fresh perspectives on customer experience. *Journal of Services Marketing* 29, 6/7 (2015), 430–435.
- [25] Wes McKinney. 2011. pandas: a foundational Python library for data analysis and statistics. *Python for High Performance and Scientific Computing* (2011), 1–9.
- [26] Negin Mehrbod, António Grilo, and Aneesh Zutshi. 2018. Caller-Agent Pairing in Call Centers Using Machine Learning Techniques with Imbalanced Data. In *2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*. IEEE, 1–6.
- [27] Fredrik Milani and Fabrizio M. Maggi. 2018. A Comparative Evaluation of Log-Based Process Performance Analysis Techniques. In *Business Information Systems*. Springer, 371–383.
- [28] Shreshtha Mundra, Anirban Sen, Manjira Sinha, Sandya Mannarswamy, Sandipan Dandapat, and Shourya Roy. 2017. Fine-Grained Emotion Detection in Contact Center Chat Utterances. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 337–349.
- [29] Eric WT Ngai, Li Xiu, and Dorothy CK Chau. 2009. Application of data mining techniques in customer relationship management: A literature review and classification. *Expert systems with applications* 36, 2 (2009), 2592–2602.
- [30] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12, Oct (2011), 2825–2830.
- [31] Leonard KM Poon, Nevin L Zhang, Tengfei Liu, and April H Liu. 2013. Model-based clustering of high-dimensional data: Variable selection versus facet determination. *International Journal of Approximate Reasoning* 54, 1 (2013), 196–215.
- [32] Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics* 20 (1987), 53–65.
- [33] Shourya Roy, Raguathan Mariappan, Sandipan Dandapat, Saurabh Srivastava, Sainyam Galhotra, and Balaji Peddamuthu. 2016. QA RT: a system for real-time holistic quality assurance for contact center dialogues. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. AAAI Press, 3768–3775.
- [34] Morteza Saberi, Anne Karduck, Omar Khadeer Hussain, and Elizabeth Chang. 2016. Challenges in Efficient Customer Recognition in Contact Centre: State-of-the-Art Survey by Focusing on Big Data Techniques Applicability. In *Intelligent Networking and Collaborative Systems (INCoS), 2016 International Conference on*. IEEE, 548–554.
- [35] Morteza Saberi, Omar Khadeer Hussain, and Elizabeth Chang. 2017. Past, present and future of contact centers: a literature review. *Business Process Management Journal* 23, 3 (2017), 574–597.
- [36] Gideon Schwarz et al. 1978. Estimating the dimension of a model. *The annals of statistics* 6, 2 (1978), 461–464.
- [37] Carlos Segura, Daniel Balcells, Martí Umbert, Javier Arias, and Jordi Luque. 2016. Automatic speech feature learning for continuous prediction of customer satisfaction in contact center phone calls. In *International Conference on Advances in Speech and Language Technologies for Iberian Languages*. Springer, 255–265.
- [38] Kah Phooi Seng and Li-Minn Ang. 2018. Video Analytics for Customer Emotion and Satisfaction at Contact Centers. *IEEE Transactions on Human-Machine Systems* 48, 3 (2018), 266–278.
- [39] Michael Waskom, Olga Botvinnik, Drew O’Kane, Paul Hobson, Saulius Lukauskas, David C Gemperline, Tom Augspurger, Yaroslav Halchenko, John B. Cole, Jordi Warmerhoven, Julian de Ruiter, Cameron Pye, Stephan Hoyer, Jake Vanderplas, Santi Villalba, Gero Kunter, Eric Quintero, Pete Bachant, Marcel Martin, Kyle Meyer, Alistair Miles, Yoav Ram, Tal Yarkoni, Mike Lee Williams, Constantine Evans, Clark Fitzgerald, Brian, Chris Fonesbeck, Antony Lee, and Adel Qalieh. 2017. mwaskom/seaborn: v0.8.1 (September 2017). (Sept. 2017). <https://doi.org/10.5281/zenodo.883859>
- [40] Hui Zou and Trevor Hastie. 2005. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67, 2 (2005), 301–320.